Speech Communication 99 (2018) 1–11

Contents lists available at ScienceDirect

Speech Communication

journal homepage: www.elsevier.com/locate/specom

Discriminative Keyword Spotting for limited-data applications

Hadas Benisty*, Itamar Katz, Koby Crammer, David Malah

Andrew and Erna Viterbi Department of Electrical Engineering, Technion - Israel, Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Keywords:
AUC
Bagging predictors
Discriminative classification
Histogram representation
Keyword Spotting
ROC
SVM

ABSTRACT

Mobile devices are widely used around the world, frequently by people speaking local languages or dialects that are not well documented. For these languages, it might not be beneficial for commercial companies to develop Automatic Speech Recognition (ASR) systems, so users of these languages cannot utilize voice activation features (often using Keyword Spotting, KWS) of their devices. Standard KWS methods aim to statistically model the generation process of the speech signal, requiring hours of recorded and transcribed speech for training, and therefore are not adequate for limited-data scenarios. In this paper we propose a new KWS method, suitable for limited-data scenarios, which can be easily applied by developers. The proposed method uses a new histogram representation for words, obtained with respect to a pre-trained Gaussian Mixture Model (GMM). Sentences are represented by fixed-length global feature vectors, extracted from the response curves obtained by a word classifier. Word and sentence classifiers are trained using a discriminative approach, which is typically robust to training-set size. The dataset for training the GMM is easy to obtain, since no annotation is required. We compared the proposed system to a Hidden Markov Model (HMM) based system, trained using the same low data-resources conditions as ours, and to a state-of-the-art ASR system, trained using either the limited data scenario, or using many hours of recorded speech. In the limited data situation, our system performs better then both benchmarks in all experiments except for clean speech of children (CSLU dataset), where it performs as good as the HMM. Since the ASR benchmark performs poorly without enough training data, we also trained it without limiting the available data. In this case the ASR benchmark performs better when tested on speech of adults (TED-LIUM dataset of TED lectures) for all noise conditions, and our system performs better when tested on speech of children with low to moderate SNR values. The results demonstrate the advantages of the proposed system, and the conditions under which it performs better.

1. Introduction

Keyword Spotting (KWS) is a task of detecting whether a specific keyword was uttered in a given speech signal. It is used, for example, in mobile applications, smart homes and security purposes. In cases where the query is given in the form of text, KWS can be viewed as a sub-task of automatic speech recognition (ASR). Some ASR systems aim at recognizing whole word terms, by using Large Vocabulary Continuous Speech Recognition (LVCSR) to generate word level transcription of the given speech signal. Other KWS systems addressing this task are based on phonetic recognizers used for ASR, thus eliminating the need for a detailed word-based language model. Still, these systems require a great amount of phonetically labeled recordings.

To illustrate a practical need for developing a low-resource KWS system, we consider the following scenario. Many smartphone users around the globe cannot utilize the voice activation properties of a device using their local language or dialect, since it is not profitable for big commercial companies to invest time and resources to obtain a labelled medium-large dataset for training an ASR system. Mobile application developers, however, may have financial interest in applying KWS for under-documented languages. In these cases, only few positive examples may be available for training. In addition, limited device computational power may also dictate low resource scenario.

In recent years developing KWS systems for under-documented languages has become a main interest in the research community. The IARPA Babel project[1] aim to foster this research: “to rapidly develop speech recognition capability for keyword search in a previously unstudied language, working with speech recorded in a variety of conditions with limited amounts of transcription.” This project has motivated many researches to examine existing and new methods for KWS onto newly collected datasets related to various, under-documented languages, such as Cantonese by Kingsbury et al. (2013), Vietnamese by Tsakalidis et al. (2014) and Chen et al. (2014), Assamese, Bengali, Haitian Creole, Lao, Zulu by Gales et al. (2014). Still, each of these

* Corresponding author.
E-mail addresses: hadasbe@techunix.technion.ac.il (H. Benisty), itamark@tx.technion.ac.il (I. Katz), koby@ee.technion.ac.il (K. Crammer), malah@ee.technion.ac.il (D. Malah).

[1] IARPA broad agency announcement IARPA-BAA-11-02, 2011.

https://doi.org/10.1016/j.specom.2018.02.003
Received 7 July 2016; Received in revised form 6 December 2017; Accepted 26 February 2018
Available online 27 February 2018

systems rely on several hours of recorded speech, along with transcription. A different approach for multilingual representation for speech recognition and KWS was also proposed in this project, but even there, at least 3 hours of transcribed recordings of the target language of interest are required (Cui et al., 2015). Although a notable effort is invested by the IARPA Babel for collecting datasets and developing recognition and spotting technologies for all spoken languages nowadays, this mission is still not completed. Many other under-documented languages and dialects spoken by millions of people (for example: Dhundari, Kinyarwanda, Ilocano, Sylheti, Chewa) are still unexplored and transcribed recordings are not available yet.

Detection of keywords for children is more challenging than for adults since their speech signals are characterized with higher variability in terms of formants location and phoneme duration as described in Gerosa et al. (2009). Existing datasets related to speech of children are very few, even for well documented languages, as they are much harder to obtain, due to privacy and parental rights. Therefore, exploring speech of children and designing recognition or KWS systems to be used by them is almost impossible for many languages, using existing methods.

Background noise and reverberations present an additional challenge for recognition and spotting systems. ASpIRE - Automatic Speech recognition In Reverberant Environments - challenge, also proposed by IARPA, is a project addressing these conditions, Harper (2015). In this project, recordings were collected by English speakers in several rooms, using one or few microphones, providing various reverberant environments. The training data consisted of about 2000 hours of transcribed speech, so systems proposed for this project were not examined in low resource environments.

In this work we present a novel discriminative method for Keyword Spotting in a limited-data environment, without the need for word- or phone-level transcription. Our method is based on two classifiers: an isolated word classifier trained using samples of the keyword and samples of non-keywords speech, and a sentence classifier trained using positive sentences (including the keyword) and negative sentences (not including it). For training the word classifier we propose a new representation for isolated words, based on a pre-trained GMM which captures the structure of the spectral feature vectors. Training a GMM requires a relatively large dataset (several hours of recorded speech, at least) to achieve sufficient statistical validity. However, GMM is typically trained using an unsupervised method - Expectation Maximization (EM), as presented by Dempster et al. (1977), and does not require any annotation. Therefore any speech recordings, relevant to the Keyword Spotting task in terms of language and/or speakers' identity or age, can be used for training the GMM.

We propose a histogram representation for keywords, based on the posterior probabilities given the pre-trained GMM. We train a discriminative binary classifier using examples of the keyword and non-keywords. We further propose a novel representation for sentences, based on the isolated keyword classifier, which produces a fixed length representation for each sentence. Using this representation we train a discriminative binary classifier for sentences.

In this paper we specifically consider the limited-data setups such as mobile applications described above, where users can be asked to record themselves only a small number of times, resulting in a very small positive dataset available for training. In such setups, where the positive training set is much smaller than the negative one, the training process may result in a classifier that is biased towards the negative class. To avoid this situation, while still exploiting the diversity of the negative training set, we use bootstrap aggregating, also referred to as bagging predictors proposed by Breiman (1996), for training the isolated word classifier, as well as for training the global classifier for sentences. According to this approach we use the majority decision of several classifiers (or predictors), each trained using the smaller positive set, and an equally sized and uniformly sampled subset of the larger negative set.

To evaluate the performance of our approach, we performed an experimental study using speech of both adults and children. We considered several challenging setups, including noisy speech signals at test time, cross-age training and testing, and various values of training set size. We used two benchmarks for comparison. The first is a Hidden Markov Model (HMM) based system which is trained using the same resources as our system, and the second is a state-of-the-art Automatic Speech Recognition (ASR) system which is trained on a large dataset of recorded speech, as well as a pronunciation dictionary and a large text corpora for language modeling. We also experimented with a low resource setup for the ASR, that is trained on a small subset of the recordings.

This paper is organized as follows. Section 2 describes previous works related to the KWS task. In Section 3, we describe our proposed approach for isolated word recognition and Keyword Spotting. Experimental results, evaluating the performance of the proposed approach compared to HMM and ASR system, are presented in Section 4. Conclusions and further research suggestions are given in Section 5.

2. Related work

Keyword Spotting using ASR was done, for example, by Garofolo et al. (2000). However, ASR systems require an enormous amount of annotated data, which is not always available for under-documented languages or speech of children, for example, Boves et al. (2009). KWS systems based on phone level recognition usually use HMM to statistically model sub-word units such as phonetic n-grams or multigrams, as presented by James and Young (1994), Thambiratnam and Sridharan (2005), Vergyri et al. (2007) and Mamou et al. (2007).

In cases where the query is given as a speech signal, Query-by-Example (QbyE) approaches are applied. These methods usually do not use language models so they require much smaller training sets and considerably less annotated data, if any. Some QbyE approaches are based on lattice representation of sub-word units, similarly to text-based systems. These supervised methods train the lattices using phonetically labelled recordings as proposed by Shen et al. (2009) and Parada et al. (2009). Unsupervised QbyE methods do not require any kind of labelled resource. Instead, they use a template representation of the keyword and compare it against a similar representation of a given speech utterance. Several methods based on a posterior representation of speech data have been proposed using various approaches: a phonetic division where the posterior values are obtained using the lattice output of a phonetic recognizer performed by Shen et al. (2009), the output of a Multi Layer perceptron (MLP) by Fousek and Hermansky (2006), statistical modeling of the speech signal using Gaussian Mixture Model (GMM) by Zhang and Glass (2009), or alternatively, using HMM as presented by Wang et al. (2011). The natural rate of speech varies with speakers and context so the posterior representation of the template and test signals usually do not match in length. Therefore most of these methods use Dynamic Time Warping (DTW). An efficient implementation for DTW has been proposed in Zhang and Glass (2011), however, using DTW still imposes a challenging computational load. In recent years Deep Neural Network (DNN) and Long Short-Term Memory Networks (LSTM) have emerged as a promising tool for signal processing and learning as proposed by Deng et al. (2013) and Hochreiter and Schmidhuber (1997). Chen et al. (2015) present a state-of-the-art QbyE method for KWS, which is robust to noise while requiring small memory footprint and low computational cost. However, it requires thousands of hours of transcribed speech and therefore cannot be used in cases of under-documented languages.

The main concern with KWS methods presented above is that they use statistical models or phonetic segmentation for classification, trained to maximize a likelihood function rather than directly maximizing the keyword detection rate. To address this issue, in recent

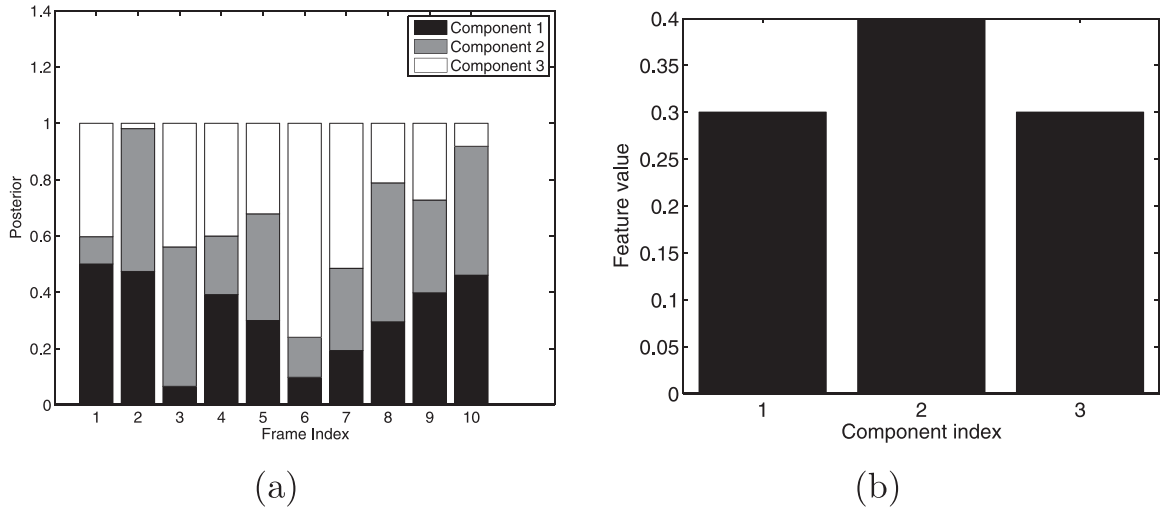

Fig. 1. A toy example: (a) The posterior probability for 10 frames and 3 mixture components, that is the probability of each component conditioned on the frame's spectral feature vector. (b) The histogram feature vector is calculated by counting the number of frames each component attains the maximal posterior, and normalizing by the total number of counts. For example, component 2 has maximal posterior in 4 out of 10 frames (frames index 2, 3, 5 and 8).

years several KWS methods have been proposed based on discriminative classification. Discriminative methods use machine learning techniques for training optimal (in terms of detection rate) binary classifiers to distinguish speech signals including a keyword from signals not including it. Keshet et al. (2009) proposed a new feature representation for speech utterances based on the estimated duration of phonemes and transition times. A linear classifier is trained using positive sentences (including the keyword) and negative sentences (not including it). This method is trained using phonetically labeled data of a medium size such as TIMIT, Garofolo et al. (1993), which consists of approximately 4 hours of recorded speech.

Two methods dealing with the case of small training set (several minutes long) have been proposed. Both methods use features extracted from the time-frequency representation of speech signals: spectro-temporal patches proposed by Ezzat and Poggio (2008) or patterns of high-energy tracks proposed by Barnwal et al. (2012). These methods use isolated utterances of the keyword, as opposed to using positive sentences as used by Keshet et al. (2009), and negative utterances including other words to train a binary classifier. Given a test sentence, a sequence of feature vectors is extracted using a sliding window. A binary classifier is then used to produce a response curve, and a final decision is taken by applying a threshold to the response curve.

3. Proposed approach

In this section we propose a new discriminative method for Keyword Spotting. This method is based on a histogram representation for classification of isolated words as described in Section 3.1, followed by global feature representation for classification of sentences, as described in Section 3.2. In Section 3.3 we describe how bagging predictors are utilized for training robust and unbiased word and sentence classifiers. An overall description of our proposed inference procedure, based on the above, is presented in Section 3.4.

3.1. Histogram representation for isolated words

Let $\mathscr{M}$ be a Gaussian Mixture Model (GMM), trained using spectral features extracted from all available training data:

$$\mathscr{M} = \{\lambda^m, \mu^m, \Sigma^m;\ m = 1,...,M\}, \tag{1}$$

where $\lambda^m \in R$, $\boldsymbol{\mu}^m \in R^P$ and $\boldsymbol{\Sigma}^m \in R^{P \times P}$ are the weight, mean vector and covariance matrix of the mth component (out of M components in the mixture), respectively, and P is the dimension of the spectral feature vectors. GMM is an unsupervised model, not requiring any labelling or other metadata, so even in cases of limited data resources such as under-documented languages, a sufficiently large amount of training data can be easily collected.

For a given word w, consider a sequence of T_w spectral feature vectors extracted from a specific utterance of w, $(\mathbf{x}_1,...,\mathbf{x}_{T_w}) \in R^{P \times T_w}$. We obtain a posteriogram representation, $\mathbf{z}_{1:T_w} = (\mathbf{z}_1,...,\mathbf{z}_{T_w}) \in R^{M \times T_w}$, with respect to the GMM, as follows:

$$z_t(m) \ = \ \mathbb{P}(m \mid \mathbf{x}_t), \quad \begin{matrix} t \ = \ 1,...,T_w \\ m \ = \ 1,...,M \end{matrix} \tag{2}$$

where $z_t(m)$ is the mth element of $\mathbf{z}_t$, and the posterior probability is given by

$$\mathbb{P}(m \mid \mathbf{x}_t) = \frac{\lambda^m \exp\{-1/2(\mathbf{x}_t - \mu^m)^\top \Sigma^{m-1}(\mathbf{x}_t - \mu^m)\}}{\sum_{n=1}^{M} \lambda^n \exp\{-1/2(\mathbf{x}_t - \mu^n)^\top \Sigma^{n-1}(\mathbf{x}_t - \mu^n)\}}. \tag{3}$$

For each vector $\mathbf{z}_t$, $t = 1,\ ...,T_w$, we set the maximal element to 1 and the rest to zero to obtain an indicator vector $\mathbf{u}_t \in R^M$ such that:

$$u_t(m) = \begin{cases} 1 & m = \underset{n=1,..,M}{\text{argmax}}\ z_t(n) \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

This means that $\mathbf{u}_t$ is an $M \times 1$ indicator of the specific Gaussian component in $\mathscr{M}$ that has the highest conditional probability, for a given spectral vector $\mathbf{x}_t$. Finally, we obtain the word histogram representation, $\mathbf{v} \in R^M$, by averaging the indicator vectors, $\mathbf{u}_{1:T_w}$, over t:

$$\mathbf{v} = \frac{1}{T_w} \sum_{t=1}^{T_w} \mathbf{u}_t. \tag{5}$$

Therefore, each element of $\mathbf{v}$ counts the fraction of times a certain Gaussian component led to the highest probability. Note that regardless of the value of T_w, the proposed histogram representation always results in an M dimensional vector (depending on the number of GMM components), thus enabling training of discriminative classification methods with fixed input dimension such as Support Vector Machine (SVM). Fig. 1 presents a toy example summarizing how to obtain a histogram representation, given a sequence of frames.

Given a positive set of histograms extracted from utterances of the keyword and a negative set extracted from utterances of non keywords, we train a binary classifier for isolated words. In the following section we use this classifier to obtain a response curve for a given sentence, which is further used for extracting a global feature vector representing the entire sentence.

3.2. Global feature representation of sentences

Given a sequence of spectral features extracted from a certain sentence (positive or negative), $\{\mathbf{x}_1, \ldots, \mathbf{x}_{T_s}\}$, we apply a sliding window of length $\alpha\overline{T}_w$, with a $\beta\overline{T}_w$ hop, where $\overline{T}_w$ is the mean length of the keyword, evaluated using the keyword utterances used for training the word classifier, and $\alpha > 1$ and $\beta < 1$ are parameters. This way, in case of a positive sentence, most of the spectral feature vectors related to the keyword would fit into at least one of the windows. For each window, we extract a histogram with respect to the GMM, $\mathcal{M}$. The sequence of histograms, $\mathbf{v}_{1:\tau_s}$, represents the sentence, where its length τ_s depends on the length of the spectral feature sequence T_s extracted from the sentence, the mean length of the keyword $\overline{T}_w$, and the sliding window hop size.

Discriminative binary classifiers usually produce a score value on which a threshold operation is applied to produce the predicted label. In case of a linear classifier this score would be the distance of the test vector from the classifying hyperplane. In case of a non-linear classifier, the score is computed through the kernel function $K(\cdot, \cdot)$:

$$S = \sum_{i=1}^{M} \alpha_i K(\mathbf{x}_i, \mathbf{x}), \tag{6}$$

where $\mathbf{x}$ is the test vector, $\mathbf{x}_1, \ldots, \mathbf{x}_M$ are the support vectors and $\alpha_1, \ldots, \alpha_M$ are their coefficients, calculated during training. Inspired by previous work of Ezzat and Poggio (2008) and Barnwal et al. (2012), we apply the word classifier trained as described in Section 3.1, to each element in the sequence of histograms. We then use the score values to form a response curve, $\mathbf{S}_{1:\tau_s} = (S_1, \ldots, S_{\tau_s})$, where S_t is the score produced by the word classifier given the tth histogram. Therefore a positive sentence is expected to yield a response curve having a distinct maximal value corresponding to the location of the keyword in the spoken sentence, while a negative sentence is expected to lead to random-like response. Fig. 2(a) and (b) present the waveform, the spectrogram and the response curve extracted from the sentences "help me unroll the new rug" and "you didn't arrive too late", respectively, for the keyword `unroll`. Note that the response curve related to the positive sentence, Fig. 2(a), has a distinct-positive valued maximum point, as opposed to the response curve related to the negative sentence, Fig. 2(b), which is quite random and mostly below zero.

A simple approach for classifying a response curve is to apply a threshold, as performed elsewhere by Ezzat and Poggio (2008) and Barnwal et al. (2012). In this paper we generalize this operation by training a binary classifier based on global features extracted from the response curve $\mathbf{S}_{1:\tau_s}$. Define σ as the standard deviation of the response curve,

$$\sigma = \sqrt{\frac{1}{\tau_s}\sum_{t=1}^{\tau_s}\left(S_t - \frac{1}{\tau_s}\sum_{t'=1}^{\tau_s} S_{t'}\right)^2}. \tag{7}$$

The global feature vector is $\phi = (M_x, m_n, a, DN)$, where

- Normalized maximal value - $M_x = \max\{\mathbf{S}_{1:\tau_s}\}/\sigma$
- Normalized minimal value - $m_n = \min\{\mathbf{S}_{1:\tau_s}\}/\sigma$
- Normalized mean value - $a = \sum_{t=1}^{\tau_s}\{\mathbf{S}_t\}/\sigma$
- Normalized dynamic range - $DN = M_x - m_n$

Given response curves extracted from positive and negative training sentences, we obtain their global feature vectors and train a binary sentence classifier.

3.3. Bagging predictors

In practice, labeled samples are harder to acquire than unlabeled ones. Therefore, we address the case where the amount of positive examples N^+ is very small, compared to the amount of negative examples N^-. It is preferable to use all available labelled data when training a discriminative classifier, to increase robustness. However, an extremely unbalanced training set will lead to a biased classifier, classifying almost everything as negative. To avoid this bias while still utilizing the variety of the negative set, we use bagging predictors as proposed by Breiman (1996). When training an isolated word classifier we randomly select negative examples from the negative set, at the same amount as the number of available positive examples, N^+. We repeat this sampling to obtain L_1 negative subsets. Each negative subset along with the positive set is used to train a binary classifier, so we end up having L_1 isolated word classifiers. We use the same strategy for training the sentence classifiers by randomly selecting L_2 negative

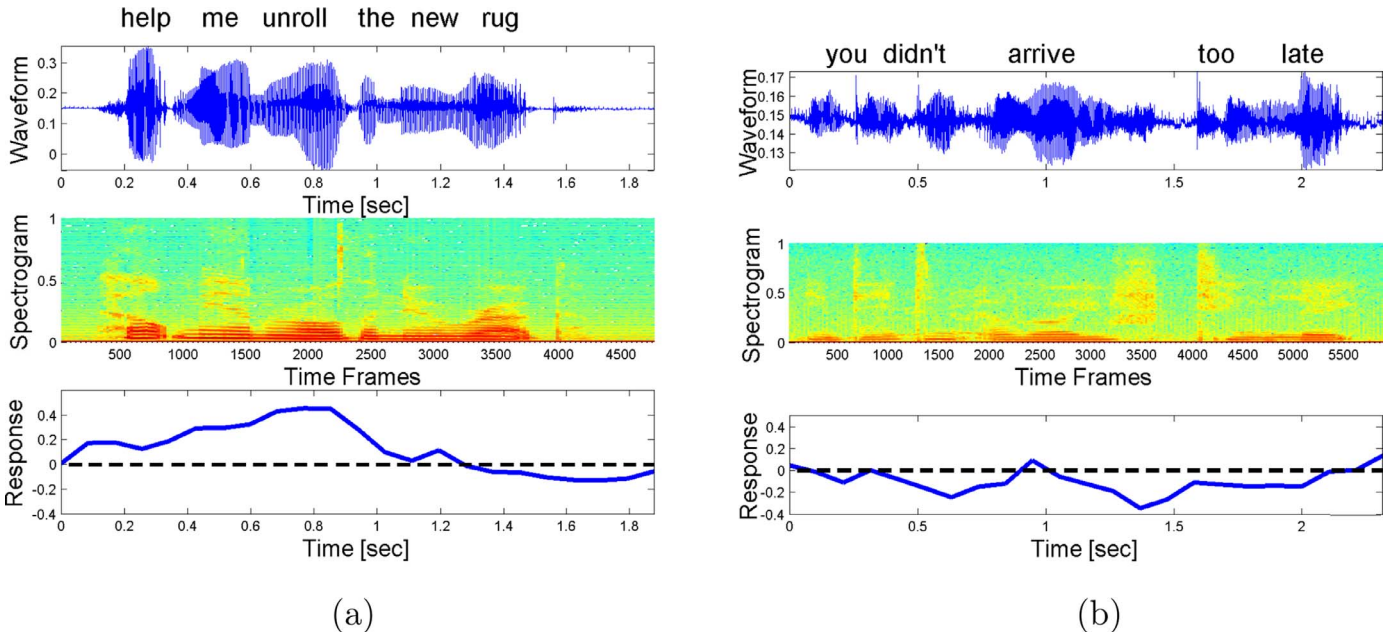

Fig. 2. Detection of the keyword `unroll` from two sentences, "help me unroll the new rug" (a) and "you didn't arrive too late" (b). Shown are the waveform (top), spectrogram (middle), and response curve and zero response (bottom, solid blue and dashed black, respectively). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

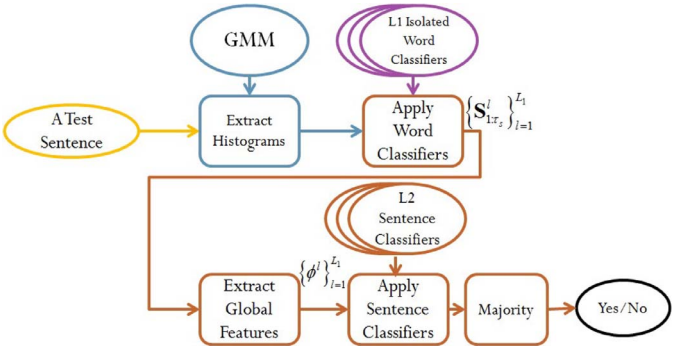

Fig. 3. Inference using the proposed approach for Keyword Spotting.

sets, each containing negative sentences at the same amount as the size of the positive set. At the end of the training process, we have L_1 isolated word classifiers and L_2 sentence classifiers.

3.4. Inference

Given a sequence of spectral feature vectors, $(\mathbf{x}_1,...,\mathbf{x}_{T_S})$, related to a test sentence, inference is made as depicted in Fig. 3. We first obtain the sequence of histograms representing the sentence, $\mathbf{v}_{1:\tau_S}$, with respect to the GMM $\mathscr{M}$, using a sliding window and Eqs. (2)–(5). L_1 isolated word classifiers are applied producing L_1 response curves $\mathbf{S}^l_{1:\tau_S}$, $l = 1, ...,L_1$. The global feature vectors, ϕ^l, $l = 1, ...,L_1$, are extracted from each response curve as described in Section 3.2. Then L_2 sentence classifiers are applied to the global feature vectors, producing $L_1 \cdot L_2$ predictions. A final decision is made by taking a majority decision.

Note that this inference process is assumed to get a sentence or a short utterance as an input. In the next section we evaluate the performance of the proposed method, along with several other benchmark systems, using datasets segmented into sentences. In case of very long utterances, or when used in an online system, an additional module is needed for slicing the speech signal into segments of about 5 s each, preferably using a speech activity detector for avoiding segments which start or end in the middle of a word.

4. Experimental study

We examined our approach in several challenging setups, using training set sizes of 5, 10 and 50 positive examples, and test sets including both clean and noisy speech signals. We used two noise types, “car” and “babble”, with SNR values ranging from 0 dB to 20 dB. To create the noisy test signals we used an available toolkit, software and noise signals, called “FaNT – Filtering and Noise Adding Tool” provided by Hirsch (2005). For clean test signals, we followed an experimental protocol similar to the one used by Ezzat and Poggio (2008) and Barnwal et al. (2012). However, we used a different dataset for reasons explained below. The features we used are Mel Frequency Cepstral Coefficients (MFCCs) along with their first and second derivatives. Features were extracted using a 25 ms window duration and 10 ms hop between frames, using Kaldi, an open source speech recognition framework described in Povey et al. (2011). For classification, we used the LIBSVM toolkit by Chang and Lin (2011), for training our proposed system for isolated word and sentence classification. Unless stated otherwise, performance was averaged over 10 random samples of the associated training set, to which we refer as ‘folds’.

4.1. Datasets

We used two datasets in our experiments. For speech of adults, we used the TED-LIUM dataset[2] of transcribed TED lectures in English, described in Rousseau et al. (2012, 2014). This dataset contains 1495 TED lectures with total duration of approximately 210 hours of recorded speech and transcripts. The dataset is split into training, development, and test sets with approximately 93,000, 500, and 1100 utterances, respectively. In addition to the recordings, the TED-LIUM corpus also includes a pronunciation dictionary and a large monolingual text corpus for language modeling, as described in Rousseau et al. (2014); Williams et al. (2015). We refer to this dataset simply as TED. For speech of children, we used recordings from CSLU, as described by Shobaki et al. (2000). This dataset consists of approximately 100 hours of recorded speech of children, aged from kindergarten to 10th grade, all American English native speakers. The recordings include isolated words, complete sentences, and spontaneous speech.

As a side note, we consider the TIMIT dataset which is commonly used in evaluation of speech algorithms and specifically in Keyword Spotting tasks. Typically, frequent words are selected to enable as large as possible dataset, as was done by Ezzat and Poggio (2008) and later by Barnwal et al. (2012). In the TIMIT dataset these keywords are `greasy`, `dark`, `wash`, and `oily`, each of which appear in approximately 640 utterances. However, each of these keywords appears in a single sentence, which is uttered by different speakers. This makes the TIMIT dataset not suitable for evaluating the performance of our algorithm. The reason is that our sentence classifier is based on features derived from a complete sentence, so the algorithm may correctly identify a keyword based on the specific sentence rather than the keyword, if this sentence appears in both training and test steps. We therefore do not report results on TIMIT.

4.2. Benchmarks

We compared our system to two benchmark systems. The first is an **Automatic Speech Recognition based KWS**, which is a state-of-the-art Automatic Speech Recognizer (ASR), using a DNN-HMM type of model, trained using the Kaldi framework and described by Veselỳ et al. (2013). Inference is done by inspection of the decoding lattice, that is a keyword is detected if the lattice contains it. We created a ROC curve by changing the size of the decoding lattice, thus controlling the tradeoff between false positives and false negatives. We trained the ASR system using different training sets, depending on the experimental context, using an existing Kaldi recipe. Training of all ASR models used the pronunciation dictionary and language model included with the TED-LIUM release. One dataset was the training set of TED. We refer to this system as TED-ASR. Another dataset was the scripted part of the CSLU training set, consisting approximately 70 hours of recorded speech. We refer to the resulting model as CSLU-ASR. Yet another dataset was a small subset of CSLU, which we considered for the low-resource scenario described in Section 1. We used approximately 25 minutes of recorded speech, equivalent to 50 positive and 100 negative sentences, which comprises the maximal training set used to train our proposed system. We refer to this system as Partial CSLU-ASR.

The second benchmark we considered is a **HMM-based KWS**. We use a similar benchmark to the one used by Keshet et al. (2009). In this approach, two HMMs are trained: a garbage + keyword model, trained using the positive sentences, and a garbage model, trained using the negative sentences. Inference is done by thresholding the likelihood ratio of the models given a sentence, and a ROC curve is created by changing the threshold value. Unlike Keshet et al. (2009) who used phonetic labeling for training their benchmark system, we used a HMM-based keyword spotter without any phone-level labeling, to allow a fair comparison with our system. A second HMM-based benchmark we used

[2] We used version 2 of the TED-LIUM corpus, which is freely available from http://www-lium.univ-lemans.fr/en/content/ted-lium-corpus.

is a GMM-HMM classifier for isolated words. In this system, all utterances of a specific word are used to train a word-level HMM. Similarly to the sentence-level HMM, no phone-level transcription was used for training. Inference is done according to the HMM with the highest likelihood score given a test word (using Viterbi decoding). Where it is not clear from context, we refer to these two benchmarks as word-level or sentence-level HMM, occasionally referring to the latter as Keyword Spotting HMM. Otherwise, we simply refer to both as HMM. The two HMM systems were trained using an available toolkit[3].

4.3. Parameter tuning

Unless stated otherwise, parameters were chosen separately for each fold, according to performance on a validation set which was randomly selected, comprising 10% of the training set size.

For the proposed system, we need to set the number of components for the GMM. We considered values from several dozens to several thousand components. In our experiments we found that the accuracy rate does not substantially improve over 500 components so this is the amount used in all the experiments. In addition, we need to set the number of word and sentence bagging predictors, L_1 and L_2. We tuned these parameters using the validation set, as follows. First, note that when we sample N sets of negative samples and train the associated N bagging models, any subset of size $M \leq N$ of the models can be used for evaluating our system using M bagging models, for every $1 \leq M \leq N$. This enables us to evaluate the average performance of many subsets of size M without the need for training more models, thus reducing the bias resulting from using a single subset. Therefore, we train $2N$ word models and $2N$ sentence models. For each candidate combination of L_1 word models and L_2 sentence models, we evaluate performance by repeated sampling of subsets of size $L_1 \times L_2$ classifiers and averaging the accuracy of the sentence prediction. For each keyword, we select the L_1 and L_2 with highest validation accuracy. In our experiments each subset was sampled 20 times, and $N = 51$, that is $L_1, L_2 \in [1, 3, 5, ..., 51]$.

For the HMM benchmark, a wide range of emitting states and mixture components was examined, 1–20 and 1–2, respectively, for each training set size and fold. The parameters range was chosen according to training set size, to avoid over-fitting. In general, as more positive examples are available for training, the tuning process results in selecting HMMs with a larger number of both emitting states and mixture components.

For the SVM classifier we considered several kernels, including the standard linear and RBF kernels, and a Chi-Square kernel defined by:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \sum_n \frac{(\mathbf{x}_i(n) - \mathbf{x}_j(n))^2}{\mathbf{x}_i(n) + \mathbf{x}_j(n)}\right), \tag{8}$$

where γ is a parameter determined by cross validation during the training stage, and $\mathbf{x}(n)$ is the nth coordinate of the vector $\mathbf{x}$. In all our experiments, the Chi-Square kernel of Eq. (8) led to the best results for isolated word classification and a linear kernel was found best for sentence classification (this combination was typically better by 10% as compared to alternatives combinations, when trained and tested using the same cross-validation process). Therefore, all results presented here were obtained accordingly.

4.4. Results

As described in Section 3, the proposed Keyword Spotting system consists of a word-level and a sentence-level classifier. To demonstrate the advantages of our approach we begin with an evaluation of the word-level classifier which we compare to a HMM-based word classifier, using the CSLU children dataset. We proceed with a Keyword Spotting task, comparing the proposed system to a HMM-based Keyword Spotting algorithm, and to a state-of-the-art ASR system using speech of adults. We conclude by studying the performance of these three systems using speech of children. All the evaluated systems were trained using clean speech, and noisy signals were used only for testing.

4.4.1. Isolated word classification - speech of children (CSLU)

We first evaluate the performance of the proposed isolated word classifier, used in our overall Keyword Spotting system. In a Keyword Spotting task, this classifier is trained for binary classification between keyword and non-keyword speech. However, here we evaluate performance on a more challenging task of multi-class classification from a given dictionary.

For training and evaluation, we used recordings of children uttering isolated words, taken from the CSLU dataset. We examined three different vocabularies, each consisting 10 words, defining three different multiclass classification tasks.[4] All parameters (number of states and components for the HMM and the SVM constant C in our approach) were tuned using 10-fold cross validation, where in each fold, 8/10 of the dataset were used for training, 1/10 for setting the parameters and 1/10 for testing. The values for number of emitting states and mixture components of the HMM were set between 6–12 and 1–2, respectively.

The spectral features of speech of children varies with age: young kids (6–10 years old) have higher variability in terms of the shape and location of formants. Towards their teens, the speech characteristics become more stable and more similar to those of adults, as described by Gerosa et al. (2009). To examine the robustness of our system and the HMM classifier to this variability, we divided the data into three age groups: “low” - kindergarten–5th grade, “high” - 6th–10th grade, and “all” - kindergarten–10th grade. We trained three classifiers using these age groups and tested each one on its corresponding group and on the other two. In order to eliminate the effect of the training set size, the “all” group was sub-sampled to match its size to the “low” and “high” groups.

The results are shown in Table 1 as the accuracy rates, mean and STD, achieved by each method and averaged over the three tasks, including all combinations of training and test sets among age groups. In general, higher accuracy is achieved when training and testing are performed using the same age group, where the “low” age group was harder for both methods due to the high variability in speech signals of young children. Nevertheless, the proposed method leads to higher accuracy rates than HMM in all cases: 4–9% higher for training and testing on the same age group and 3–13% higher for cross-age training and testing. Also note that the STD of the HMM classifier is between 1.3 and 3.4 for the “low” and “high” age groups while the STD of the proposed system is lower than 1 in both cases. For the “all” age group both methods lead to similar and low STD values. This indicates that the proposed classifier is more robust to training set size and variability than the HMM classifier, as it leads to more consistent accuracy rates.

4.4.2. Keyword Spotting - speech of adults (TED lectures)

We now evaluate performance of the proposed KWS system and benchmarks for clean and noisy speech of adults using the TED dataset. We chose 62 words which appear at least 40 times in the test set,[5]

[3] http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html.

[4] The three vocabularies used for the isolated word classification experiments are: (1) background, bathe, behind, beyond, bigfoot, biology, birthmark, boomerang, breath, bronco. (2) earthquake, easier, eight, employees, endure, engrave, ethnic, explosion, faithful, fancy. (3) gumshoe, handshake, hardship, hawthorne, herbalist, homemaking, hoof, hopeful, hourly, humor.

[5] The chosen keywords are: about, actually, because, been, by, don’t, from, get, go, going, good, had, has, he, here, how, i’m, into, just, know, like, make, me, more, much, my, no, now, one, out, people, really, right, say, see, self, some, something, than, that’s, them, then, there, these, thing, things, think, time, two, up, us, very, want, way, we’re, well, when, who, will, world, would, years.

Table 1
Isolated word classification of speech of children (CSLU dataset): classification accuracy rates, mean and STD values, averaged over three different 10-words vocabularies.

Train data		Test data		
		"Low" [%]	"High" [%]	"All" [%]
"Low"	HMM	89.0 ± 1.3	85.0 ± 2.9	86.0 ± 1.3
	proposed	**94.6 ± 0.5**	**91.1 ± 0.2**	**93.2 ± 0.3**
"High"	HMM	75.0 ± 2.7	91.0 ± 3.4	79.0 ± 4.2
	proposed	**86.1 ± 0.8**	**97.2 ± 0.6**	**90.5 ± 0.7**
"All"	HMM	81.7 ± 0.2	85.7 ± 0.3	83 ± 0.1
	proposed	**94.0 ± 0.1**	**95.9 ± 0.2**	**94.0 ± 0.1**

omitting common words that appear too frequently.[6]

To demonstrate the influence of the amount of positive examples available to the algorithm, we trained the examined systems using several sets of different sizes, consisting of 5, 10 and 50 positive examples, where for each set 90% of samples were taken from the training set and used for training, and 10% taken from the development set and were used for validation, that is for tuning the parameters of the examined methods. All test set samples containing keywords were used for testing, with an average of 64 utterances per keyword. In all the experiments, a single negative set was used, consisting of 100 sentences which do not include any of the keywords.

For training our system, single keywords were extracted using the available TED-LIUM transcription and automatic alignment using Kaldi. Note, however, that single keywords can be recorded in isolation so that in principle sentence-level transcription is not needed for our system.

As described in Section 3.3, in order to increase the robustness of the system and deal with highly unbalanced training sets (in terms of the number of positive and negative samples), especially in the case of 5 and 10 positive examples, we use bagging predictors for word and sentence classification, and take their average vote as the overall classification result. The number of bagging predictors for the word and sentence classifiers, L_1 and L_2, was set using the validation set, as described in Section 4.3.

Clean and noisy speech. We compared the performance of the proposed approach and the benchmark systems under clean and noisy conditions. We trained each system using a training set consisting clean speech and applied them to clean and noisy versions of the test set. Results in this section were averaged over detection of the 62 keywords listed in the beginning of this section. For our system and the HMM benchmark, performance was averaged over 10 random draws of the training set, for each keyword and training set size. The TED-ASR system was trained once on the full TED-LIUM training set (in Section 4.4.3 we also consider training of the ASR system on smaller datasets).

Figs. 4 and 5 present the AUC obtained by the proposed system and the two benchmarks, averaged over all keywords, where two types of noise were added to the speech signals. The two noise types were "babble" and "car", at several SNR values, ranging from 0 dB to 20 dB. For the HMM benchmark and the proposed system, 95% confidence interval for the mean was obtained considering a multi-sample average of random variables, one for each keyword, with different means and unknown variances, since we expect each keyword to have different mean performance. For the TED-ASR benchmark system, the performance of each keyword is based on a size-one sample (single training set). Therefore, we do not show confidence interval for the TED-ASR mean performance. As expected, the TED-ASR system leads to the best results, as it uses more extensive resources for training, compared to the other two examined systems. Compared to the HMM benchmark system (both using the same resources), the proposed system has a distinct advantage for clean and noisy speech, at all SNR values and for both noise types.

Sensitivity to bagging parameters. In addition, in order to examine the sensitivity of our algorithm to a specific choice of the bagging parameters, we calculated for each keyword the validation accuracy for each L_1, L_2 combination, relative to the maximal validation accuracy for that keyword, and averaged over keywords. Formally, if r_{ijk} is the validation accuracy for keyword k with $L_1 = i$ and $L_2 = j$, then the relative average accuracy is given by

$$\bar{r}_{ij} = \frac{1}{K}\sum_{k=1}^{K}\frac{r_{ijk}}{\max\limits_{i,j} r_{ijk}},$$

where K is the number of keywords. Ideally, $\bar{r}_{jk}$ should be close to 1 and should depend on j, k only weakly, at least for some subset of values. Fig. 6 shows $\bar{r}_{jk}$ for a training set of size 5. It can be seen that as long as L_1 and L_2 are not too small, the average validation accuracy is within 98% or more of the accuracy achieved by the best choice of L_1 and L_2. This demonstrates that our system is not very sensitive to the specific choice of the bagging parameters. Similar results were achieved for training sets of size 10 and 50. Nevertheless, in the following experiments we did tune L_1 and L_2 in order to allow for a fair comparison with the HMM benchmark, and considering that the computational overhead is not high, as explained in Section 4.3.

4.4.3. Keyword Spotting - speech of children (CSLU)

We evaluated the performance of our proposed approach also for speech of children taken from the CSLU dataset. In this experiment we used unified training and test sets consisting of all age groups together. As before, training was repeated 10 times using randomly sampled training sets. At each repetition, 8/10 of the set were used for training, 1/10 of the set for parameters tuning, and 1/10 for testing in noisy conditions. The bagging parameters L_1 and L_2 were tuned as described in Section 4.3. We chose words which appear at least 50 times, which is the maximal positive set we used for training. This resulted in only four words: `bathe`, `one`, `two` and `unroll`, used as keywords in this section.

Comparison with HMM on clean speech. Table 2 presents the AUC (mean and confidence interval) obtained by the HMM benchmark and the proposed system for all age groups (kindergarten–tenth grade). The AUC was averaged over detection of the four keywords and over 10 repetitions of each experiment, using randomly selected training sets. The proposed method has a distinct advantage for 5 and 10 positive training examples, whereas for 50 the benchmark system is better.

Choosing the ASR training set. As described in Section 4.2, we considered three options for the ASR training set. In order to choose the most appropriate one, we evaluated the performance of the ASR system on KWS for speech of children, using these three training sets, which we termed TED-ASR, CSLU-ASR, and Partial CSLU-ASR. For clean speech, The TED-ASR benchmark leads to AUC = 0.73 ± 0.01, and the CSLU-ASR leads to AUC = 0.93 ± 0.001. Both system were trained using many hours of speech, and used the same language resources (pronunciation dictionary and language model), so this deterioration is apparently caused by the domain difference, that is training and testing on different domains (adults and children in the case of TED-ASR), compared to a single domain used both for training and testing (children in the case of CSLU-ASR. When using Partial CSLU-ASR, the accuracy rates were close to random prediction, indicating that the training set size is too small. We next evaluated the performance of the three ASR benchmark variants on noisy speech signals of children. Fig. 7 shows the AUC values obtained by the ASR

[6] The **omitted** frequent words are: `the, and, to, of, a, that, in, is, you, I, this, it, we, so, for, but, have, on, are, was, with, what, they, it's, can, be, all, at, not, as, do, if, an, or, our`.

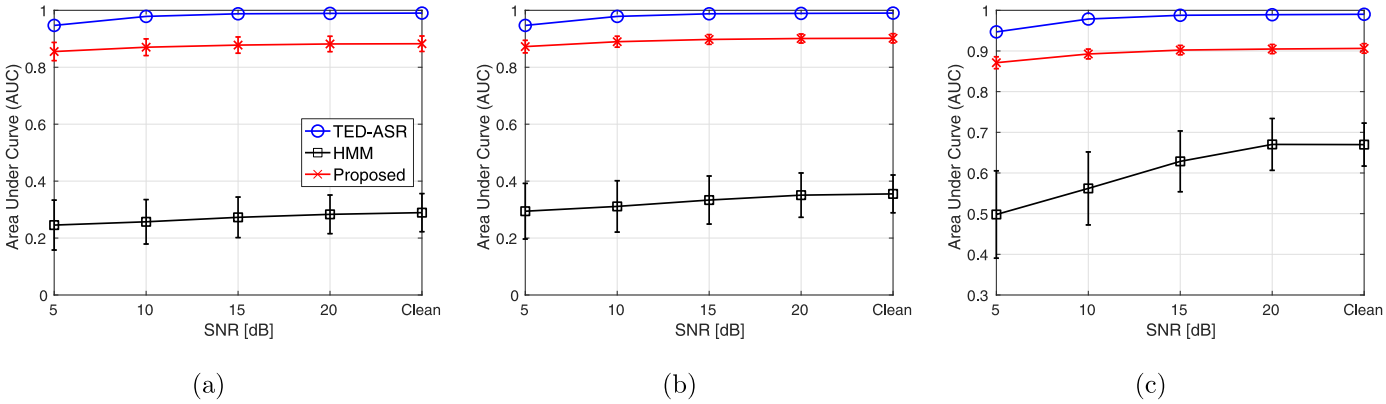

Fig. 4. AUC averaged over detection of 62 keywords taken from TED lectures speech of adults, tested on clean and noisy speech (**babble noise**). The systems are TED-ASR (blue circle), HMM (black square), and proposed approach (red X). The HMM and the proposed system were trained using 5, 10 and 50 positive training sentences, (a), (b), and (c), respectively, each repeated 10 times on randomly selected training sets. The TED-ASR system was trained on the complete training set of TED-LIUM corpus. For the HMM and proposed system we also show 95% confidence intervals for the mean, *magnified* $\times$ 20 *for better visualization*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

benchmark system, trained using the three datasets and tested on noisy versions of CSLU using "car" and "babble" noise types. Also shown are results for clean speech, as given in Table 2. Firstly, it is clear that 25 minutes of recorded speech are not sufficient for proper training of the ASR system as its performance is equivalent to random prediction. Secondly, the differences between speech of adults and children is again well demonstrated, as training the system on speech of adults deteriorates performance. Therefore, for the rest of this section we use CSLU-ASR as a benchmark, bearing in mind that it requires much more resources compared to the proposed system and the HMM-based benchmark.

Figs. 8 and 9 show the AUC, averaged over detection of the four keywords, obtained by the HMM benchmark, the CSLU-ASR benchmark, and the proposed system. When testing on clean signals, our system leads to higher AUC values than the HMM benchmark when 5 or 10 positive examples are available for training whereas for 50 positive examples, both systems perform the same. When testing on noisy signals, our proposed system is more robust: it outperforms the HMM-benchmark system at all SNR values, for both noise types and for all training set sizes. Comparing to the CSLU-ASR benchmark, below some SNR threshold the proposed system is more robust to noise and performs better. This threshold is higher as more training samples are available: for 5 and 10 training samples, this threshold is 10db and 15db, respectively, while for 50 training examples, only on clean speech does the CSLU-ASR performs better.

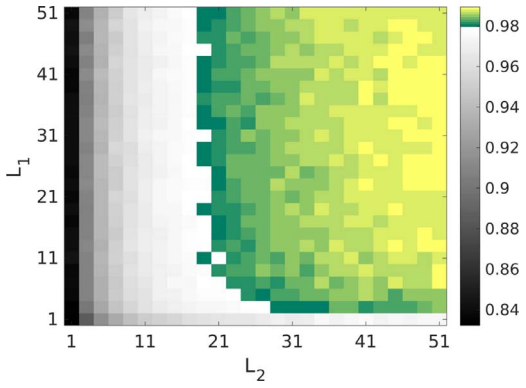

Fig. 6. Relative validation accuracy as a function of bagging parameters L_1 and L_2, measured relatively to the maximal accuracy per keyword, and averaged over all keywords. Dataset is TED speech of adults, and training set size is 5 positive examples. Relative accuracy below 98% is shown in hues of gray, above 98% shown in hues of green. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.5. Limitations of the proposed approach

So far we have demonstrated the advantages of the proposed method, which enables training of a KWS system even at extreme

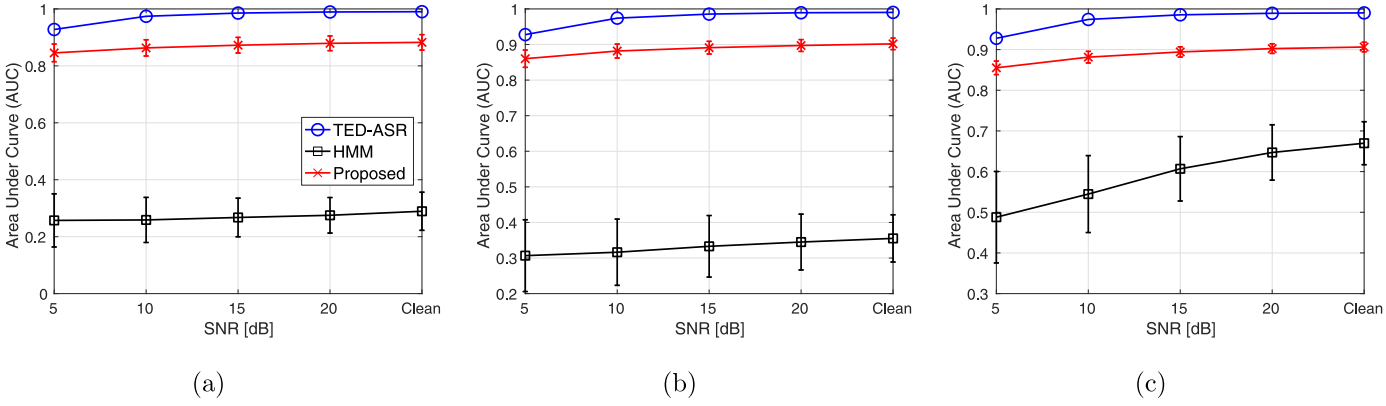

Fig. 5. AUC averaged over detection of 62 keywords taken from TED lectures speech of adults, tested on clean and noisy speech (**car noise**). The systems are TED-ASR (blue circle), HMM (black square), and proposed approach (red X). The HMM and proposed system were trained using 5, 10 and 50 positive training sentences, (a), (b), and (c), respectively, each repeated 10 times on randomly selected training sets. The TED-ASR system was trained on the complete training set of TED-LIUM corpus. For the HMM and for the proposed system we also show 95% confidence intervals for the mean, *magnified* $\times$ 20 *for better visualization*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Mean AUC results obtained by the HMM-based KWS and the proposed system applied to speech of children (CSLU). An average was taken over detection of four words: `bathe`, `one`, `two` and `unroll`, and over 10 repetitions of the experiment per word, using randomly selected training sets.

		# Positive examples		
	Method	**5**	**10**	**50**
Mean AUC	**HMM**	0.5 ± 0.1	0.6 ± 0.1	**0.9 ± 0.1**
	Proposed	**0.6 ± 0.05**	**0.7 ± 0.05**	0.8 ± 0.1

situations of limited-data resources. In this section, we discuss the limitations of the proposed method, and compare its performance to the state-of-the-art TED-ASR system. Surely, in cases of under-documented languages and/or speech of children, when dozens of hours of transcribed recordings are unavailable, proper training of a state-of-the-art ASR system is impossible. Still, we use it as a skyline for this section to discuss several important issues. The results in this section relate to clean speech of adults from the TED-LIUM dataset. The proposed system was trained using 50 positive and 100 negative examples, and the ASR system was trained using the complete dataset, both as presented in Section 4.4.2. The main limitations of the proposed system are:

- **New Keywords** – the proposed system is based on discriminative classifiers, each trained for a specific keyword. Therefore, to add a new keyword, additional training is needed, as opposed to an ASR system, which does not require additional training.
- **Left/Right Context** – In some applications, the left/right context of the keyword might be important, for example “Google OK” vs. “OK Google”. Using the proposed approach would lead to similar representation of the two phrases, which may lead to detection errors. An ASR system, however, would probably manage to distinguish between the two.
- **Substrings** – a keyword which is a substring of a non-keyword, for example `out` and `about`. In these cases, the histograms of the substring (keyword) could be very similar to the histogram obtained by sliding over the uttered sentence including the longer string (non-keyword). The ASR system, however, is more robust to these situations since it is trained for recognizing sub-word units and also relies on a full language model. A closely related case is when a non-keyword is a substring of a keyword. Table 3 presents four examples of substrings, their appearances in the test set, and the false positive rates of both systems. The ASR has a distinct advantage as its false positive rates are much lower than the ones obtained by the proposed method.
- **Acoustically Confusable Words** – two different words which are pronounced similarly can lead to false detections. Table 4 presents the false positive rates of two examples. The proposed method falsely detected the keyword in all the examined cases, whereas the ASR system did not.

To conclude, when having many hours of transcribed recordings of speech for training, it is best to use a state-of-the-art ASR system, except for speech of children at low to moderate SNR values. For limited training resources, the proposed system is preferable, also compared to the HMM benchmark system tested, under both clean and noisy speech conditions.

5. Conclusion

In this paper we propose a novel approach for Keyword Spotting, specifically adequate for limited-data setups, such as mobile applications, under-documented languages, and speech signals of children. We propose fixed-length representations for words and for sentences, enabling the training of discriminative classification methods such as Support Vector Machine (SVM). We avoid bias in training by using bootstrap aggregating, also referred to as bagging predictors, where a series of classifiers are trained using randomly sampled subsets of the larger training set. Experimental study demonstrated the advantages of the proposed method on speech of both adults and children, in several challenging setups, considering small training-set sizes and different background noises - “car” and “babble”. We compared the performance of the proposed system to two benchmark systems: a HMM Keyword Spotting system, and a state-of-the-art ASR system based on the Kaldi framework.

In the situation of having about two hundred hours of transcribed recorded speech for training, the ASR system leads to the best results for adults. However, the accuracy rates significantly deteriorate when training on speech of adults and testing on speech of children; training the ASR system on speech of children leads to a significant improvement (provided that enough data is available for training). Nevertheless, for speech of children, on all but clean speech and using just 50 positive examples, the proposed system leads to as good as or higher accuracy rates than the ASR system, trained using the complete corpus. Compared to the HMM benchmark, our system is significantly better when tested on speech of adults for all training set sizes and noise

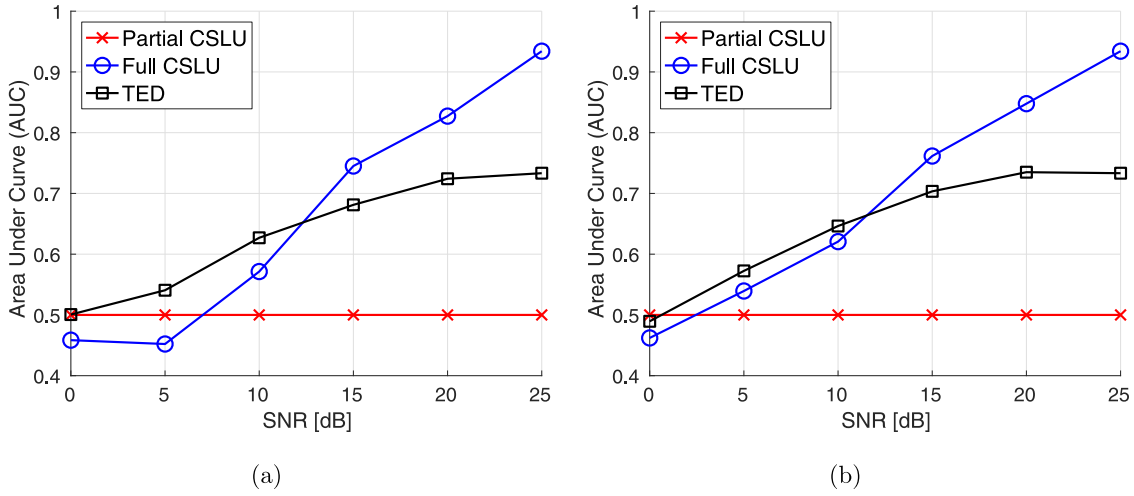

Fig. 7. Performance of the ASR system, tested on children and trained using adults TED dataset (black squares), the full CSLU dataset, consisting approximately 70 hours of speech (blue circles), and partial CSLU dataset, approximately 25 minutes of speech, or 50 positive and 100 negative sentences (red asterisk). AUC was averaged over four keywords, `bathe`, `one`, `two`, and `unroll`. Test sentences were taken from the “All” age group (kindergarten to tenth grade) of CSLU. Noise types are “car” (a) and “babble” (b). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

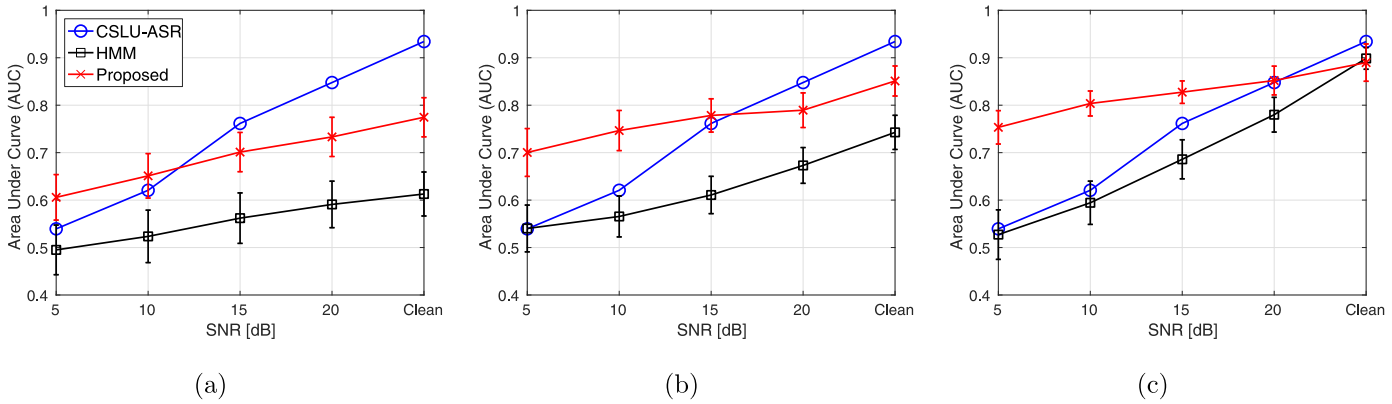

Fig. 8. AUC averaged over detection of four keywords, `bathe`, `one`, `two`, and `unroll`, taken from the "All" age group (kindergarten to tenth grade) of CSLU, tested on clean and noisy speech (**babble noise**). The systems are CSLU-ASR (blue circles), HMM (black squares), and proposed approach (red X). The HMM and proposed system were trained using 5, 10 and 50 positive training sentences, (a), (b), and (c), respectively, each repeated 10 times on randomly selected training sets. The CSLU-ASR system was trained on the complete training set of CSLU corpus. For the HMM and for the proposed system we also show 95% confidence intervals for the mean (not magnified). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

levels. For speech of children, our system performs better than the HMM for all noise levels when using small training sets (of size 5 or 10), while for a set size of 50, it is better for all noise levels, but for clean speech, for which our system and the HMM benchmark show comparable performance.

As for *further research*: since the histogram representation of keywords presented in this paper is obtained with respect to a GMM, the temporal correspondence of the spectral feature vectors is ignored. An alternative model, considering the temporal context of spectral feature vectors, such as DNN, could provide better modeling of keywords, and as a result, improve the detection rate. In this work we proposed a set of global features for representing sentences. These features were selected since they characterise the differences between positive and negative response curves. Still, exploring other features may lead to improved representation and classification of positive and negative response curves and therefore to improved detection rate.

The proposed approach relies on a histogram representation which leads to robust classifier, even in low data-resource conditions. However, this representation also causes some limitations such as training of new keywords, left/right context, substrings, and confusable words. A simple solution for some of these issues, based on the proposed system, could be to expand the histogram representation to two histograms - one to the first half of the keyword, and another to the second half, so the final representation for the keyword would be a

Table 3
False positive rate of **substrings** for the proposed system and the ASR system, using the TED-LIUM dataset. The 'Appearances' field is the number of non-keyword sentences in the test set.

Keyword, other string	Appearances	ASR	Proposed method
`out, about`	125	0.4	0.83
`about, out`	78	0.18	0.63
`some, something`	55	0.36	0.95
`something, some`	49	0.14	0.9

Table 4
False positive rate of **acoustically confusable words** for the proposed system and the ASR system, using the TED-LIUM dataset. The 'Appearances' field is the number of non-keyword sentences in the test set.

Keyword, other string	Appearances	ASR	Proposed method
`thing, think`	72	0.32	0.81
`them, then`	52	0.4	0.62

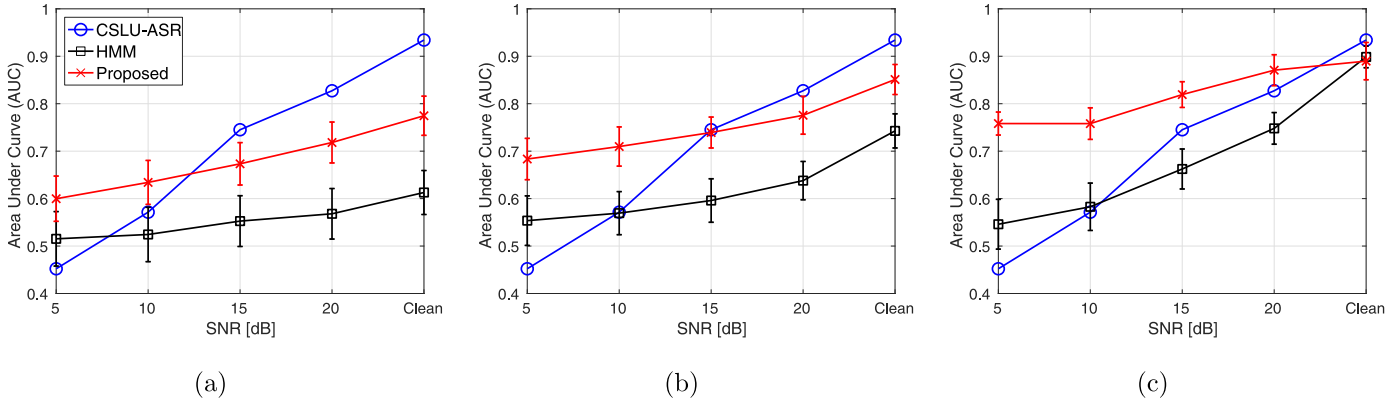

Fig. 9. AUC averaged over detection of four keywords, `bathe`, `one`, `two`, and `unroll`, taken from the "All" age group (kindergarten to tenth grade) of CSLU, tested on clean and noisy speech (**car noise**). The systems are CSLU-ASR (blue circles), HMM (black squares), and proposed approach (red X). The HMM and proposed system were trained using 5, 10 and 50 positive training sentences, (a), (b), and (c), respectively, each repeated 10 times on randomly selected training sets. The CSLU-ASR system was trained on the complete training set of CSLU corpus. For the HMM and for the proposed system we also show 95% confidence intervals for the mean (not magnified). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

concatenation of the two histograms. More research is needed to explore this solution and examine its performance.

Acknowledgments

This research was funded in part by the Ministry of Israeli Economics grant 50360 and in part by an Israeli Science Foundation grant ISF- 1567/10. The authors thank K. Adam and S. Mousazadeh, from Linguistech Ltd, for their support and useful discussions in the course of the work, and to the devoted SIPL staff: Nimrod Peleg, Yair Moshe, Ziva Avni and Avi Rosen for their technical support.

References

Barnwal, S., Sahni, K., Singh, R., Raj, B., 2012. Spectrographic seam patterns for discriminative word spotting. Proceedings of ICASSP. IEEE, pp. 4725–4728.
Boves, L., Carlson, R., Hinrichs, E.W., House, D., Krauwer, S., Lemnitzer, L., Vainio, M., Wittenburg, P., 2009. Resources for speech research: present and future infrastructure needs. Proceedings of INTERSPEECH. pp. 1803–1806.
Breiman, L., 1996. Bagging predictors. Mach. Learn. 24 (2), 123–140.
Chang, C.C., Lin, C.J., 2011. LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. 2, 1–27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
Chen, G., Parada, C., Sainath, T.N., 2015. Query-by-example keyword spotting using long short-term memory networks. Proceedings of ICASSP. IEEE, pp. 5236–5240.
Chen, N.F., Sivadas, S., Lim, B.P., Ngo, H.G., Xu, H., Ma, B., Li, H., et al., 2014. Strategies for Vietnamese keyword search. Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, pp. 4121–4125.
Cui, J., Kingsbury, B., Ramabhadran, B., Sethy, A., Audhkhasi, K., Cui, X., Kislal, E., Mangu, L., Nussbaum-Thom, M., Picheny, M., et al., 2015. Multilingual representations for low resource speech recognition and keyword search. Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on. IEEE, pp. 259–266.
Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. B 39, 1–38.
Deng, L., Hinton, G., Kingsbury, B., 2013. New types of deep neural network learning for speech recognition and related applications: an overview. Proceedings of ICASP. IEEE, pp. 8599–8603.
Ezzat, T., Poggio, T., 2008. Discriminative word-spotting using ordered spectro-temporal patch features. Proceedings of INTERSPEECH. pp. 35–40.
Fousek, P., Hermansky, H., 2006. Towards ASR based on hierarchical posterior-based keyword recognition. Proceedings of ICASSP. 1. IEEE, pp. I.
Gales, M.J., Knill, K.M., Ragni, A., Rath, S.P., 2014. Speech recognition and keyword spotting for low-resource languages: BABEL project research at CUED. SLTU. pp. 16–23.
Garofolo, J.S., Auzanne, C.G., Voorhees, E.M., 2000. The TREC spoken document retrieval track: a success story. 1, 1–20 LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.
Garofolo, J.S., Consortium, L.D., et al., 1993. TIMIT: Acoustic-Phonetic Continuous Speech Corpus. Linguistic Data Consortium.
Gerosa, M., Giuliani, D., Narayanan, S., Potamianos, A., 2009. A review of ASR technologies for children's speech. Proceedings of WOCCI. ACM, pp. 7.
Harper, M., 2015. The automatic speech recognition in reverberant environments (ASpIRE) challenge. Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on. IEEE, pp. 547–554.
Hirsch, H.-G., 2005. FaNT-Filtering and Noise Adding tool. ACM Trans. Intell. Syst. Technol. Software available at http://dnt.kr.hs-niederrhein.de/index964b.html?option=com_content&view=article&id=22&Itemid=15&lang=de.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.
James, D.A., Young, S.J., 1994. A fast lattice-based approach to vocabulary independent wordspotting. Proceedings of ICASSP. 1. IEEE, pp. I–377.
Keshet, J., Grangier, D., Bengio, S., 2009. Discriminative keyword spotting. Speech Commun. 51 (4), 317–329.
Kingsbury, B., Cui, J., Cui, X., Gales, M.J., Knill, K., Mamou, J., Mangu, L., Nolden, D., Picheny, M., Ramabhadran, B., et al., 2013. A high-performance cantonese keyword search system. Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, pp. 8277–8281.
Mamou, J., Ramabhadran, B., Siohan, O., 2007. Vocabulary independent spoken term detection. Proceedings of SIGIR. ACM, pp. 615–622.
Parada, C., Sethy, A., Ramabhadran, B., 2009. Query-by-example spoken term detection for OOV terms. Proceedings of ASRU. IEEE, pp. 404–409.
Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., Vesely, K., 2011. The Kaldi speech recognition toolkit. IEEE 2011 Workshop on Automatic Speech Recognition and Understanding. IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.
Rousseau, A., Deléglise, P., Esteve, Y., 2012. TED-LIUM: an automatic speech recognition dedicated corpus. LREC. pp. 125–129.
Rousseau, A., Deléglise, P., Estève, Y., 2014. Enhancing the TED-LIUM corpus with selected data for language modeling and more ted talks. LREC. pp. 3935–3939.
Shen, W., White, C.M., Hazen, T.J., 2009. A Comparison of Query-by-Example Methods for Spoken Term Detection. Technical Report. DTIC Document.
Shobaki, K., Hosom, J.-P., Cole, R., 2000. The OGI kids' speech corpus and recognizers. Proceedings of ICSLP, Beijing, China.
Thambiratnam, K., Sridharan, S., 2005. Dynamic match phone-lattice searches for very fast and accurate unrestricted vocabulary keyword spotting. Proceedings of ICASSP. pp. 465–468.
Tsakalidis, S., Hsiao, R., Karakos, D., Ng, T., Ranjan, S., Saikumar, G., Zhang, L., Nguyen, L., Schwartz, R., Makhoul, J., 2014. The 2013 BBN vietnamese telephone speech keyword spotting system. Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, pp. 7829–7833.
Vergyri, D., Shafran, I., Stolcke, A., Gadde, V.R.R., Akbacak, M., Roark, B., Wang, W., 2007. The SRI/OGI 2006 spoken term detection system. Proc. INTERSPEECH. pp. 2393–2396.
Veselỳ, K., Ghoshal, A., Burget, L., Povey, D., 2013. Sequence-discriminative training of deep neural networks. INTERSPEECH. pp. 2345–2349.
Wang, H., Lee, T., Leung, C.-C., 2011. Unsupervised spoken term detection with acoustic segment model. Speech Database and Assessments (Oriental COCOSDA). IEEE, pp. 106–111.
Williams, W., Prasad, N., Mrva, D., Ash, T., Robinson, T., 2015. Scaling recurrent neural network language models. Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on. IEEE, pp. 5391–5395.
Zhang, Y., Glass, J.R., 2009. Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams. Proceeedings of ASRU. IEEE, pp. 398–403.
Zhang, Y., Glass, J.R., 2011. An inner-product lower-bound estimate for dynamic time warping. Proceedings of ICASSP. IEEE, pp. 5660–5663.